AMENDMENTS TO THE CLAIMS

Applicant submits below a complete listing of the current claims, including marked-up claims with insertions indicated by underlining and deletions indicated by strikeouts and/or double bracketing. This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of the Claims

1. (Currently Amended) A processing system for accessing data, the processing system comprising:

a processor for executing instructions;

a stream register unit being part of the processor and configured ~~connected~~ to supply a first type of data to the processor, the first type of data being data supplied from a peripheral, the stream register unit including at least one stream register unit FIFO configured to store the first type of data received from the peripheral;

a FIFO ~~connected~~ coupled to the peripheral to receive said first type of data from the peripheral and connected to the stream register unit by a communication path, along which said first type of data can be supplied from the FIFO to the at least one stream register unit FIFO; and

a memory bus, separate from the communication path, connected between a data memory and the processor, across which the processor can access a second type of data, the second type of data being randomly accessible data held in the data memory.

2. (Canceled)

3. (Original) A processing system according to claim 1, wherein data is supplied from the FIFO to the stream register unit in accordance with requests for data made by the processor to the stream register unit and forwarded to the FIFO.

4. (Original) A processing system according to claim 3, wherein the said requests are made as accesses to volatile variables.

5. (Original) A processing system according to claim 3, wherein the FIFO is arranged to, upon receiving a request for data from the stream register unit, send a signal to the stream register unit indicating availability of the requested data.

6. (Original) A processing system according to claim 5, wherein if the FIFO contains the requested data, the said signal to the stream register unit indicates that the data is available, and the FIFO is further arranged to send a signal to the stream register unit comprising the data.

7. (Original) A processing system according to claim 6, wherein the stream register unit is arranged to, following receipt of the signal comprising the data, supply the data to the processor.

8. (Original) A processing system according to claim 6, wherein the stream register unit is arranged to, following receipt of the signal comprising the data, send a signal to the FIFO indicating that it has taken the data.

9. (Original) A processing system according to claim 8, wherein the said signal to the FIFO further indicates the next location in the FIFO from which data is required.

10. (Original) A processing system according to claim 5, wherein the FIFO is further arranged to, if it does not contain the requested data, send a different signal to the stream register unit indicating that the data is not available.

11. (Original) A processing system according to claim 10, wherein the stream register unit is arranged to, if the signal sent by the FIFO is the said different signal indicating that the data

is not available, send a stall signal to the processor, causing the processor to stop executing instructions.

12. (Original) A processing system according to claim 10, wherein the FIFO is further arranged to, if following sending of the said different signal to the stream register unit the data subsequently becomes available, send a signal to the stream register unit indicating that the data is available, and to send a signal comprising the data to the stream register unit.

13. (Currently amended) A processing system according to claim 5, further comprising a timeout generator, arranged for communication with the processor and the stream register unit, and arranged to, if the signal sent by the FIFO is ~~the said different~~ a signal indicating that the data is not available, after a predetermined period of time, send a timeout signal to the processor, causing the processor to interrupt such that it can execute other instructions.

14. (Original) A processing system according to claim 13, wherein if following sending of the timeout signal to the processor the data subsequently becomes available, the timeout generator is arranged to receive a signal instructing it to cease sending the timeout signal, and to, upon receipt of the said instruction, cease sending the timeout signal.

15. (Original) A processing system according to claim 13, wherein the stream register unit is arranged to, if following sending of the timeout signal to the processor the data subsequently becomes available, send the data to the processor.

16. (Original) A processing system according to claim 1, wherein the stream register unit is associated with a register file containing a plurality of registers and a load/store unit arranged to receive data from the stream register unit and temporarily store the data in the register file.

17. (Original) A processing system according to claim 16, wherein the processor is arranged to retrieve data from the register file.

18. (Original) A processing system according to claim 1, wherein data is supplied from the FIFO to the stream register unit in accordance with requests for data made by the processor to the stream register unit and forwarded to the FIFO, wherein the stream register unit is associated with a register file containing a plurality of registers and a load/store unit arranged to receive data from the stream register unit and temporarily store the data in the register file, wherein the processor is further arranged to make requests for data to the stream register unit via the load/store unit.

19. (Original) A processing system according to claim 1, wherein the stream register unit comprises one or more FIFOs connected to receive data from the FIFO connected to the stream register and supply the data to the processor.

20. (Original) A processing system according to claim 3, wherein the request for data is a request for a single data item.

21. (Original) A processing system according to claim 1, further comprising one or more additional FIFOs linked together between the said FIFO and the communication channel.

22. (Original) A processing system according to claim 1, wherein the data from the peripheral is video data.

23. (Previously presented) A processing system according to claim 22, wherein the peripheral is a video processing system.

24. (Currently amended) A streaming data handling system, comprising:
a processor;

a stream register ~~associated with~~ being part of the processor and configured to supply data from a peripheral to the processor, the stream register including at least one stream register FIFO configured to store the data received from the peripheral; and

a FIFO memory coupled to the peripheral to receive the data from the peripheral and connected to the least one stream register FIFO ~~the processor~~ via a communication path ~~the stream register~~,

wherein the stream register and the FIFO operate the same data handling protocol such that the stream register can receive streamed data items from the FIFO memory and supply them to the processor in the received order.

25. (Currently amended) A system according to claim 24, further comprising a timeout generator, arranged for communication with the processor and the stream register, and arranged to, if the signal sent by the FIFO is a signal indicating that the data is not available, after a predetermined period of time, send a timeout signal to the processor, causing the processor to interrupt such that it can execute other instructions.

26. (Currently amended) A stream register connectable between a processor and a peripheral, the stream register being part of the processor and comprising:

a receiver arranged to receive a request for a data item from the processor;

at least one FIFO configured to store the data item received from the peripheral; and

a stream engine, arranged to:

receive the request for the data item;

determine whether the requested data item is in the at least one FIFO;

if the requested data item is not in the at least one FIFO, send a stall signal to the processor; and

send the request to the peripheral and receive one or more signals back from the peripheral indicating availability of the requested data item, and, if the data item is available, send the data item to the processor, and, if the data item is not available and if the stall signal has been

active for a predetermined amount of time, send a timeout signal to the processor causing the processor to interrupt such that it can execute tasks other than the request.

27. (Original) A stream register according to claim 26, wherein the stream engine is arranged to the interrupt signal to the processor after a predetermined period of time.

28. (Original) A stream register according to claim 27, wherein the stream engine is further arranged to, if the data is available, temporarily store the data in a register file for access by the processor.

29. (Original) A stream register according to claim 27, wherein the stream engine is further arranged to, following sending of the timeout signal to the processor, if the data item subsequently becomes available, receive a signal instructing it to cease sending the timeout signal, and to, upon receipt of the said instruction, cease sending the timeout signal to the processor and temporarily store the data in a register file for access by the processor.

30. (Currently amended) A stream register, being part of a processor, connectable between [[a]] the processor and a memory, the stream register comprising:

a receiver arranged to receive a request for a data item from the processor;

at least one FIFO configured to store the data item received from the peripheral; and

a stream engine, arranged to:

receive the request for the data item;

determine whether the requested data item is in the at least one FIFO;

if the requested data item is not in the at least one FIFO, send a stall signal to the processor; and

send the request to the memory and receive one or more signals back from the memory indicating availability of the requested data item, and, if the data item is available, send the data item to the processor, and, if the data item is not available and if the stall signal has been active for a

predetermined amount of time, send a timeout ~~stall~~ signal to the processor causing the processor to interrupt such that it can execute tasks other than the request.